La distribution Anaconda

Environnement de développement et accès aux packages

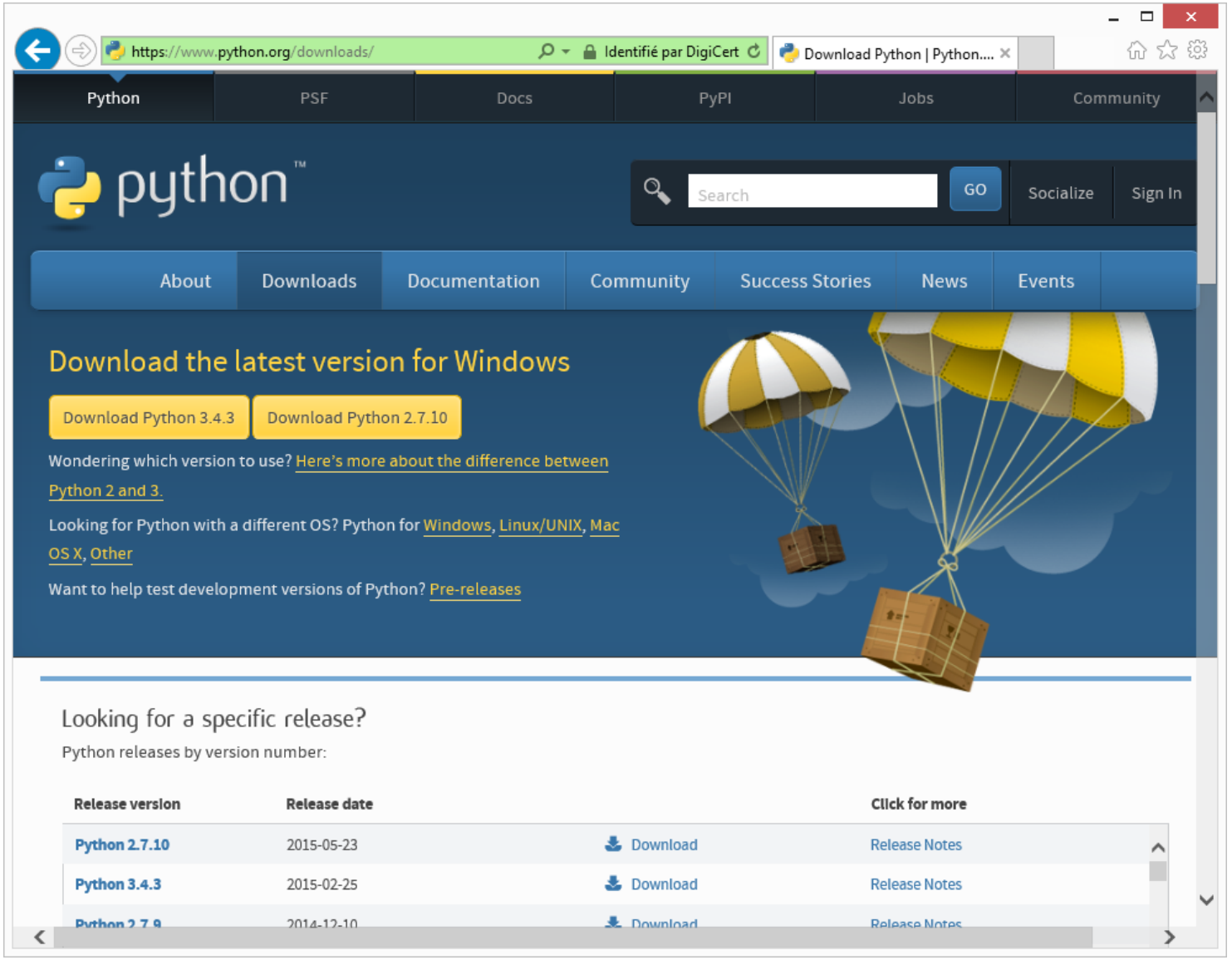
Ricco Rakotomalala

http://eric.univ-lyon2.fr/~ricco/cours/cours_programmation_python.html

Environnement de travail et gestion des packages

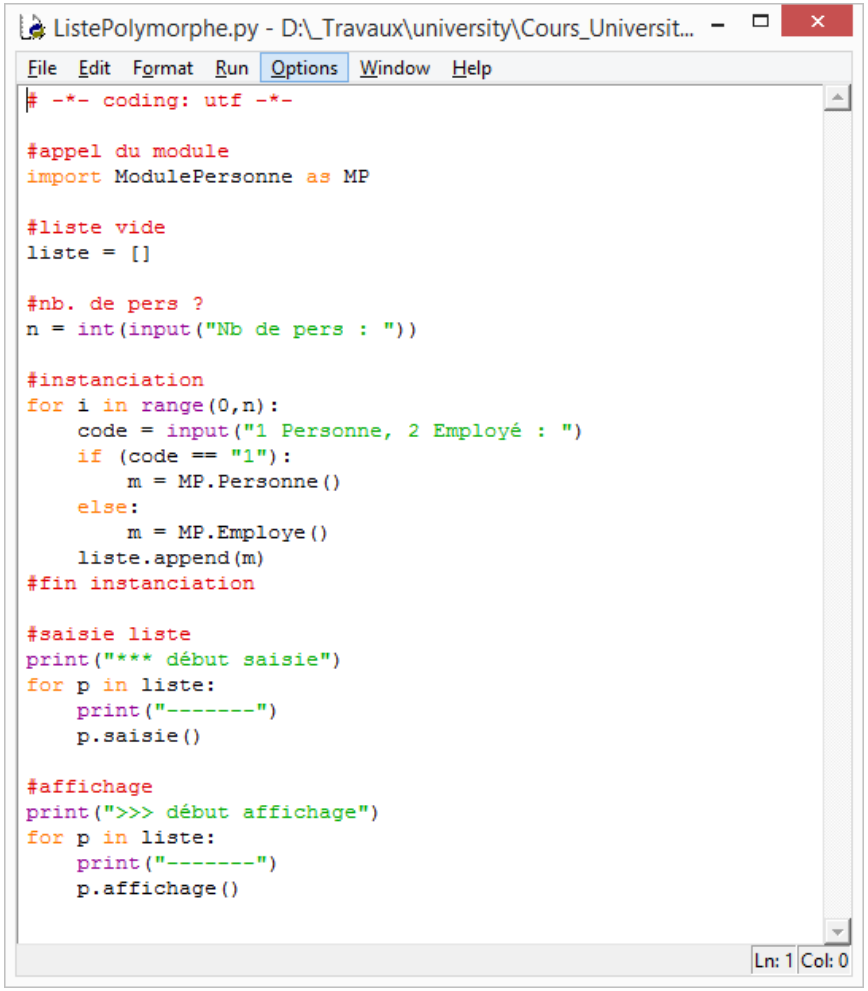
DISTRIBUTION STANDARD DE PYTHON

La distribution standard de Python

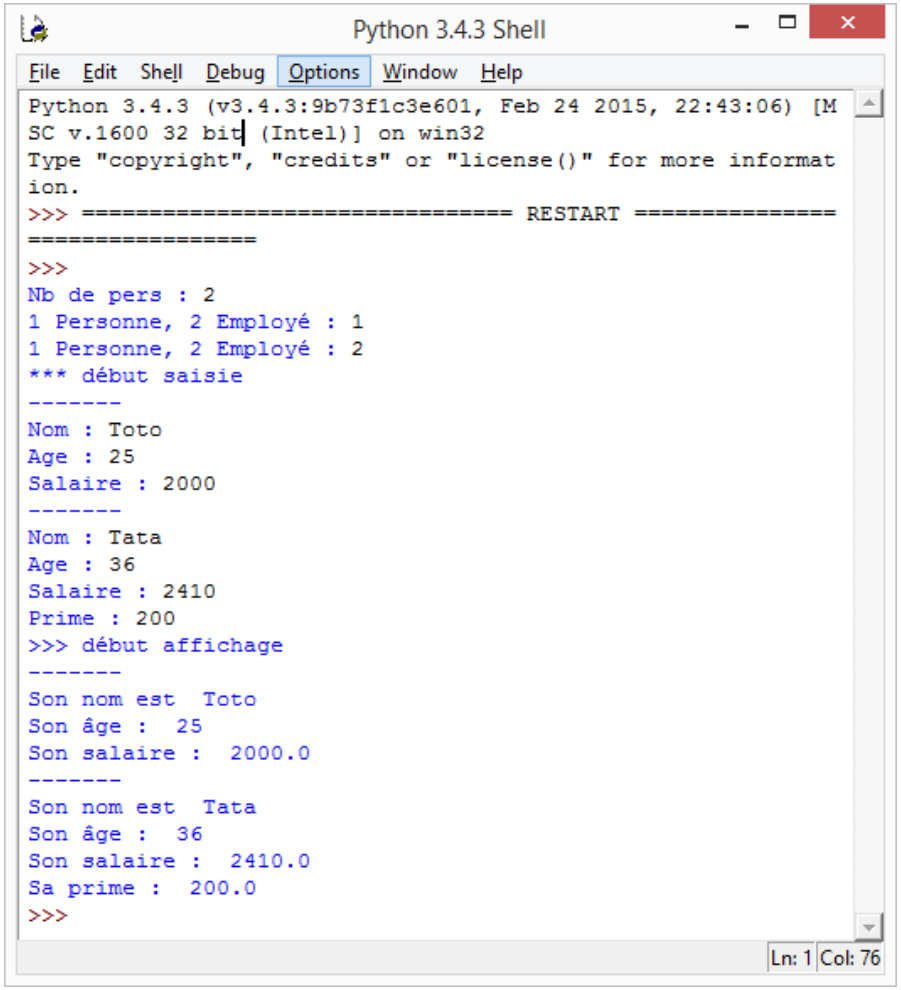


Le site de Python met à notre disposition les distributions officielles

Environnement de travail standard

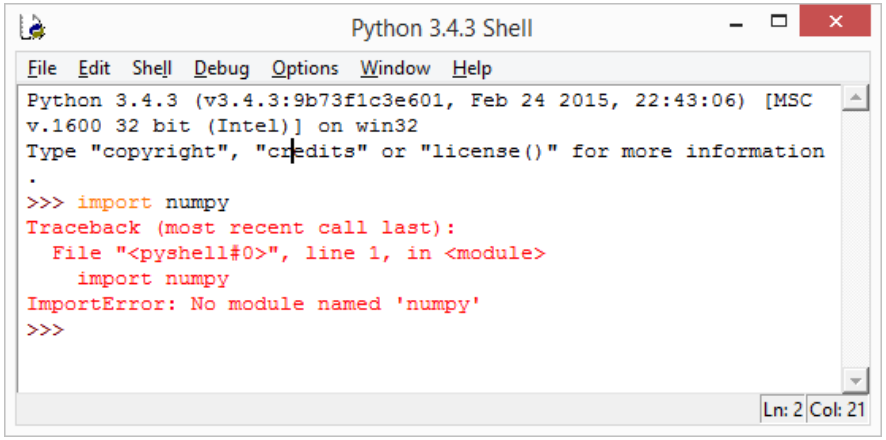


IDLE, environnement de développement



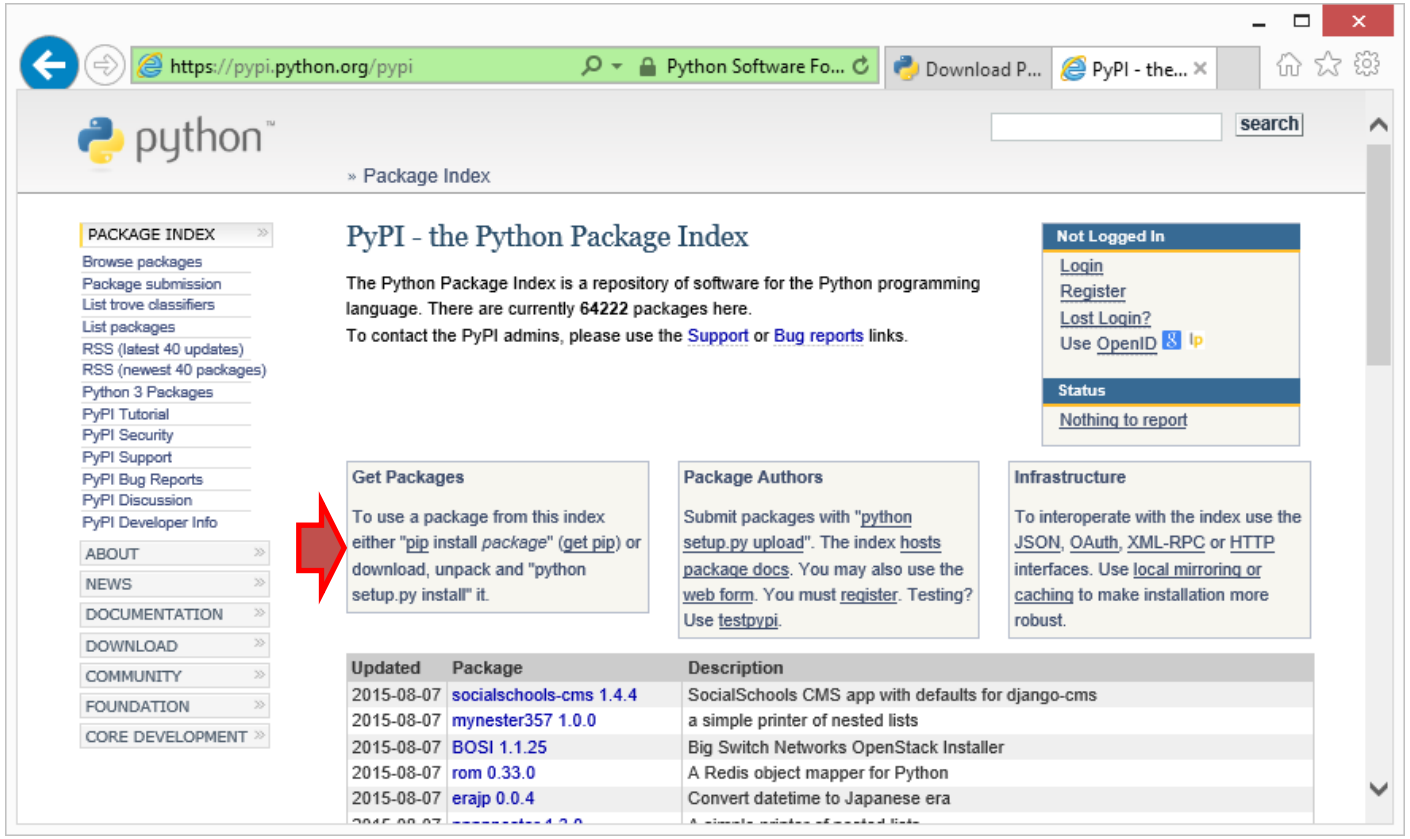
Python Shell, pour l'exécution des programmes. Fonctionnement interactif possible.

Problème des packages

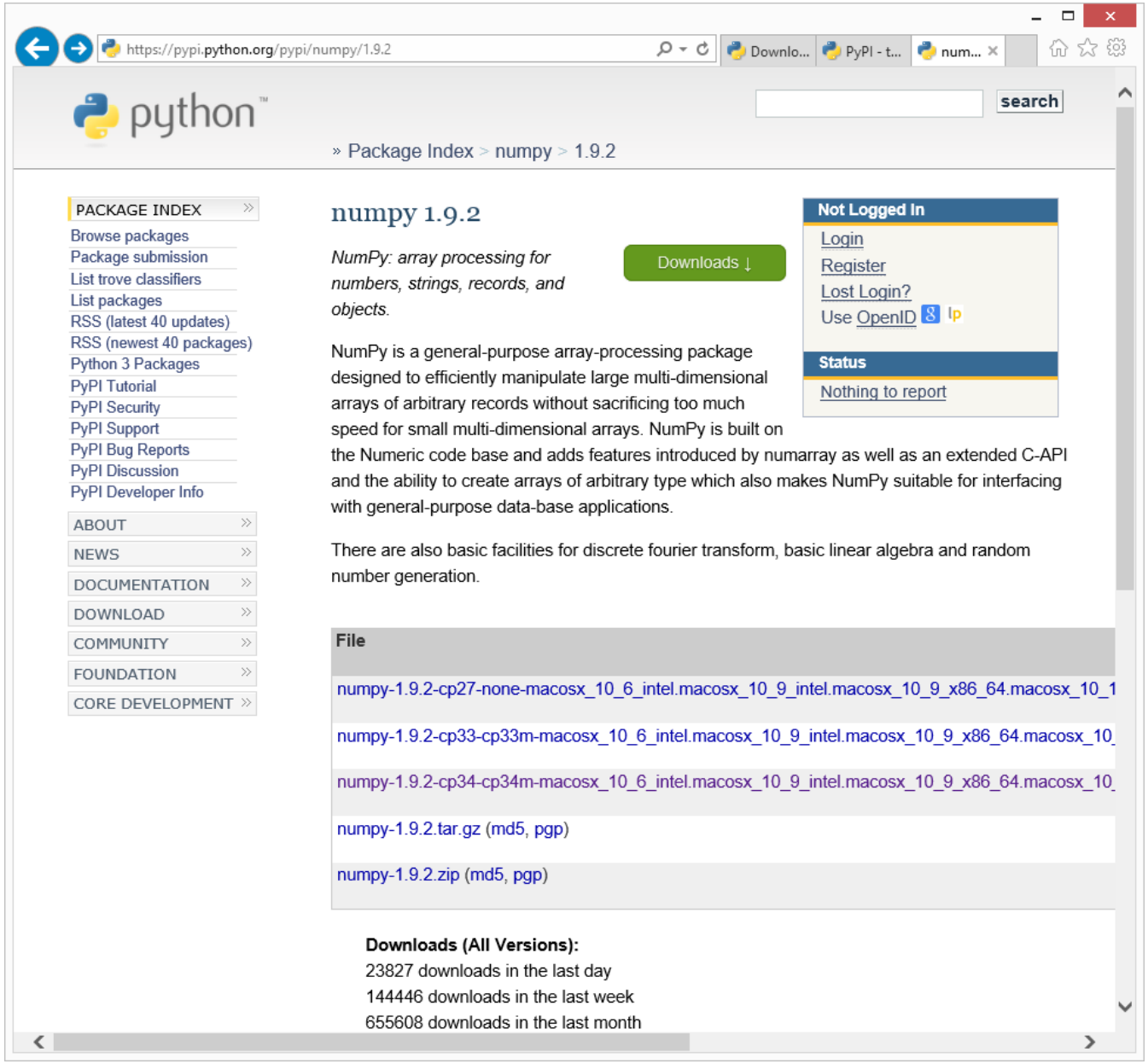


L'affaire se corse lorsqu'il s'agit d'installer des modules additionnels (ex. numpy)

Il faut récupérer le package sur PyPI et procéder à l'installation avec le logiciel PIP. L'opération n'est pas évidente du tout parce que la documentation n'est pas claire, notamment concernant les dépendances.

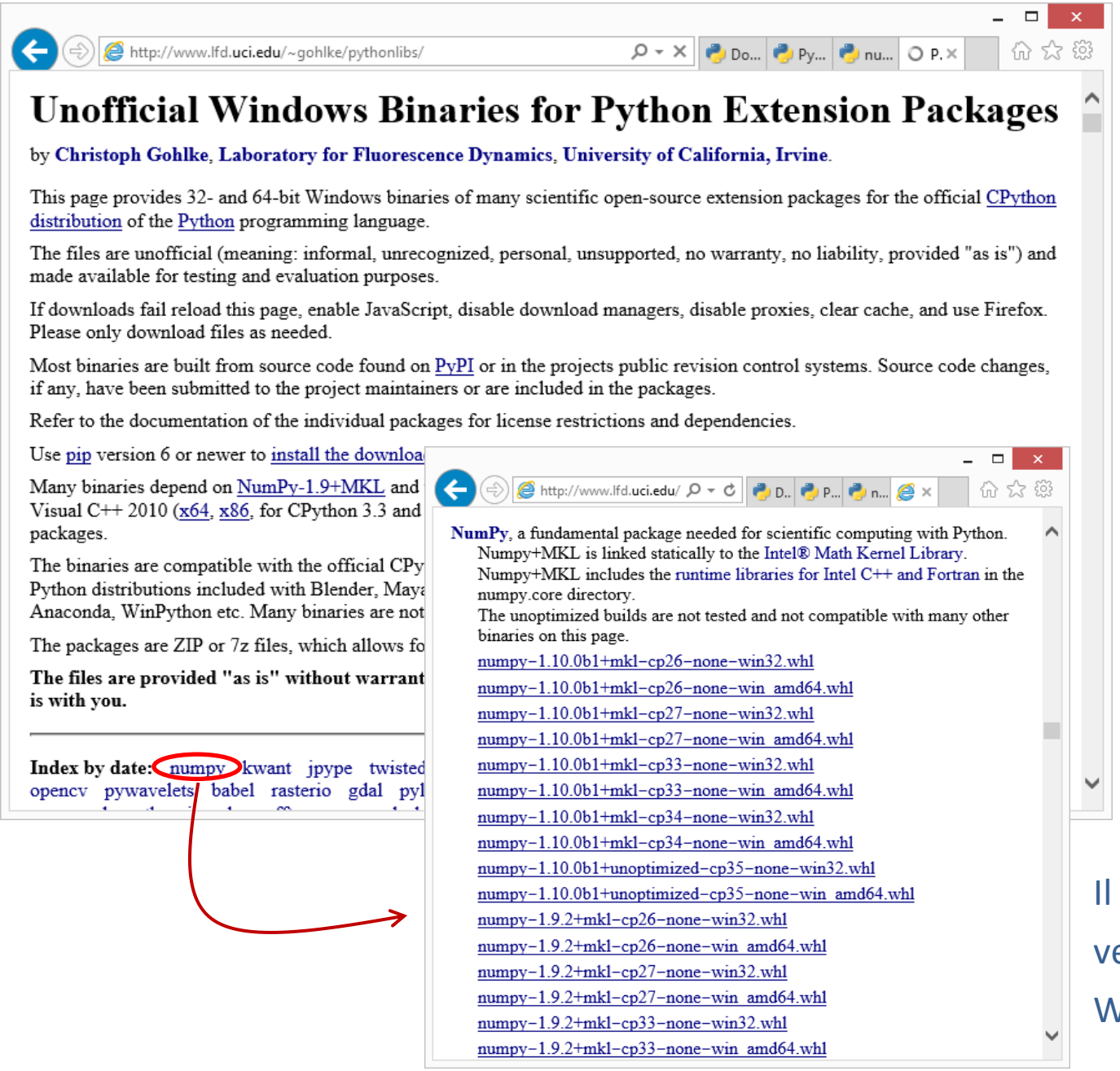


Exemple pour « numpy »



Comment procéder sous Windows ? On a un peu (beaucoup) du mal à s'y retrouver.

Le format « Wheel »



Un fichier « Wheel » (.whl) est une archive compilée permettant de d'installer un package. Pour Windows, les packages sont accessibles sur un site non officiel (???).

Il faut choisir les bonnes versions de Python et de Windows, puis utiliser PIP.

Installation d'un fichier « .whl » avec PIP

http://www.maths-algo.fr/algo/cours/cours_pyhon_installer.htm

2) Installer avec pip les packages indispensables à un environnement de travail scientifique

Il y a quatre packages indispensables: numpy, scipy, matplotlib et pillow. Python intègre depuis la version 3.4 un outil de gestion des packages, pip (pour package installation python), dont l'exécutable est situé dans C:\Python34\Scripts.
En cas de besoin, consulter le site officiel de pip. Voilà comment l'utiliser avec un PC sous Windows et connecté à l'internet:

(1) Télécharger sur le site de Christoph Gohlke (ou récupérer sur une clé usb) les fichiers .whl des packages désirés correspondant à la version de Python. Avec Python 3.4 et en mai 2015, ces fichiers s'appellent:

- numpy-1.9.2+mkl-cp34-none-win_amd64.whl
- scipy-0.15.1-cp34-none-win_amd64.whl
- Pillow-2.8.1-cp34-none-win_amd64.whl
- matplotlib-1.4.3-cp34-none-win_amd64.whl

(2) Recopier ces fichiers dans le répertoire C:\Python34\Scripts
(3) Créer un fichier batch (d'extension .bat) contenant les lignes suivantes, qui doivent être adaptées si les noms ont changés

```
cd C:\Python34\Scripts
pip install numpy-1.9.2+mkl-cp34-none-win_amd64.whl
pip install scipy-0.15.1-cp34-none-win_amd64.whl
pip install Pillow-2.8.1-cp34-none-win_amd64.whl
pip install matplotlib-1.4.3-cp34-none-win_amd64.whl
pause
```

(4) Exécuter ce fichier batch: les packages sont installés. De plus d'autres petits packages nécessaires sont installés automatiquement (c'est pour cette raison que l'ordinateur doit être connecté pendant l'exécution du batch).
Si l'ordinateur n'est pas connecté à l'internet lors de l'exécution du batch, il faudra que votre batch contiennent les lignes suivantes et que les fichiers supplémentaires .whl aient été auparavant récupérés et recopiés dans le répertoire C:\Python34\Scripts.

```
cd C:\Python34\Scripts
pip install numpy-1.9.2+mkl-cp34-none-win_amd64.whl
pip install scipy-0.15.1-cp34-none-win_amd64.whl
pip install Pillow-2.8.1-cp34-none-win_amd64.whl
pip install pytz-2015.2-py2.py3-none-any.whl
pip install pyparsing-2.0.3-py3-none-any.whl
pip install python_dateutil-2.4.2-py2.py3-none-any.whl
pip install six-1.9.0-py2.py3-none-any.whl
pip install matplotlib-1.4.3-cp34-none-win_amd64.whl
pause
```

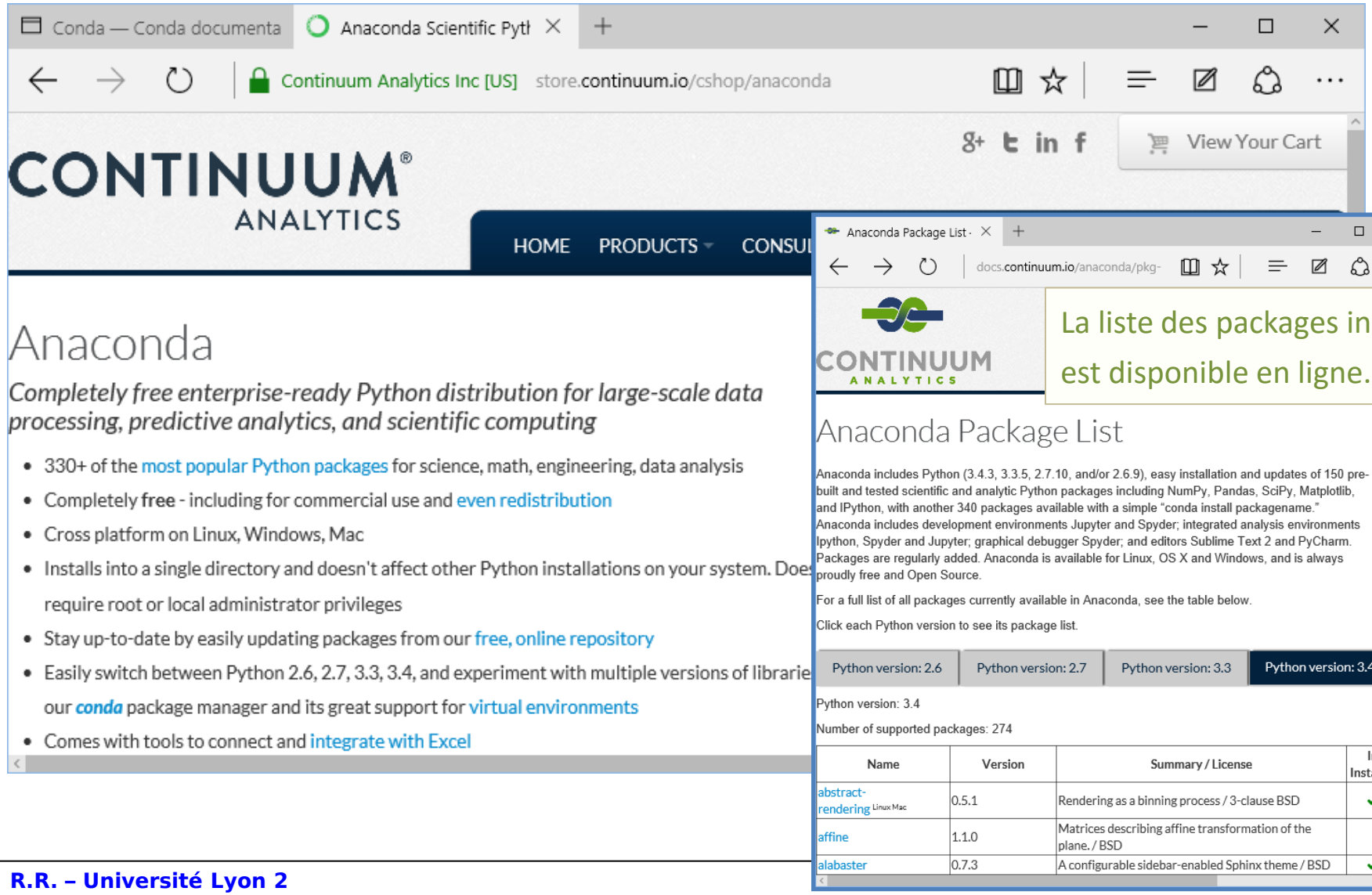

Un des très rares sites qui explique de manière « simple » et directement reproductible la procédure d'installation des packages (les librairies scientifiques ici).

Il faut trouver une solution moins contraignante, au moins pour les enseignements.

Intégration directe des « principaux » package et nouvel environnement de travail

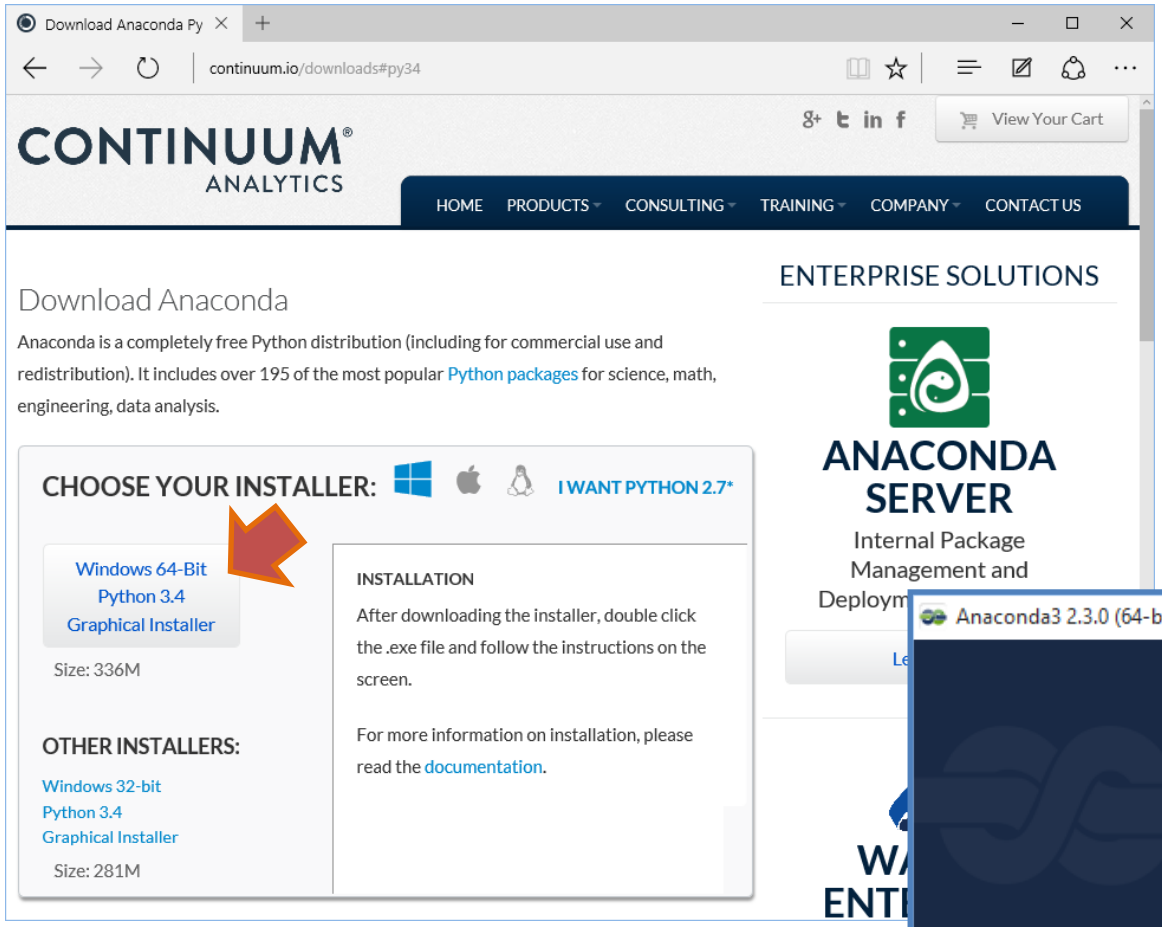
LA DISTRIBUTION ANACONDA

ANACONDA est une distribution Python libre qui intègre directement un grand nombre de packages (il n'est donc plus nécessaire de les installer, mais on peut en ajouter d'autres si nécessaire avec le gestionnaire de packages Conda).

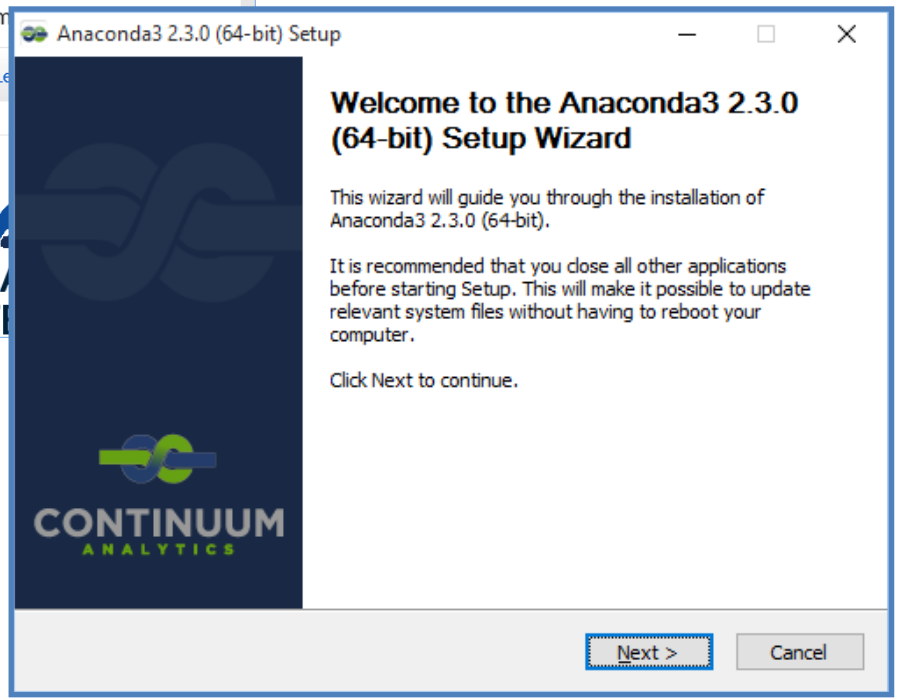


La liste des packages installés est disponible en ligne.

Installation de Anaconda



Charger la version qui vous convient.



Et lancer le setup

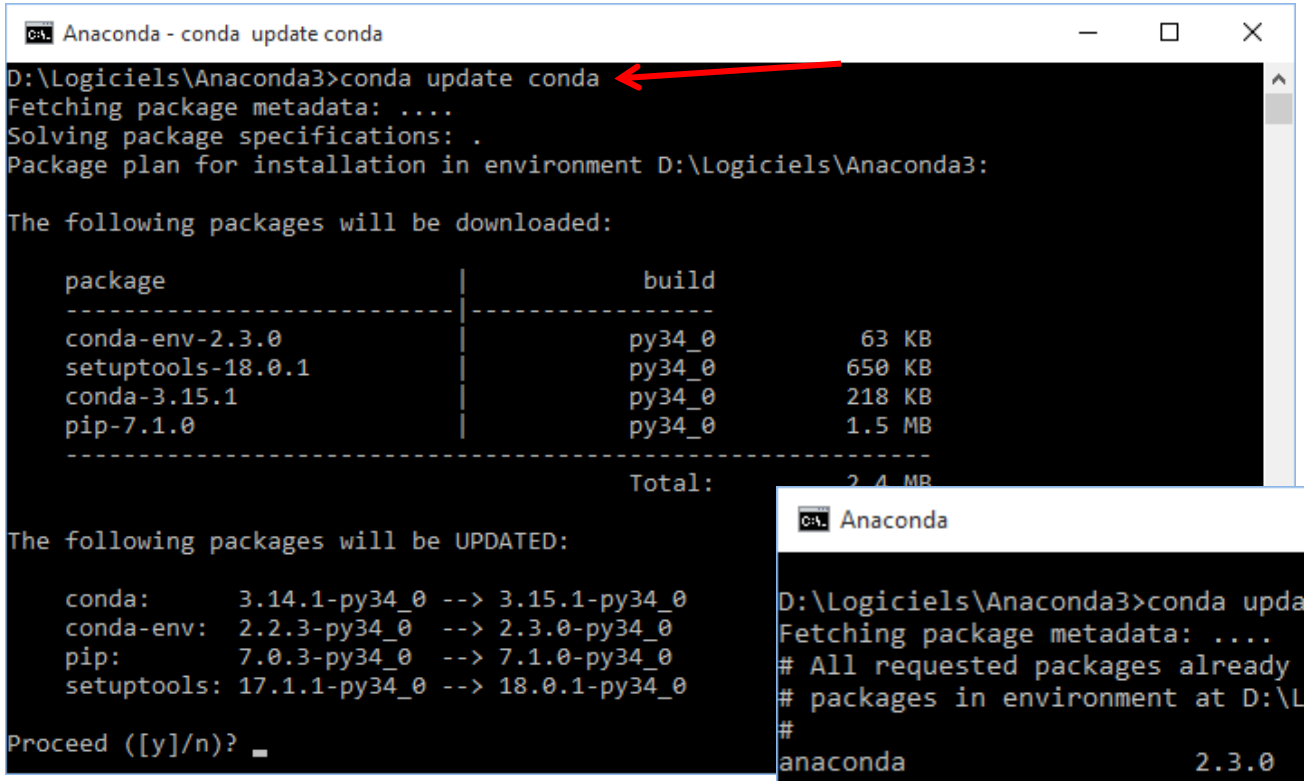
Disposer des dernières mises à jour

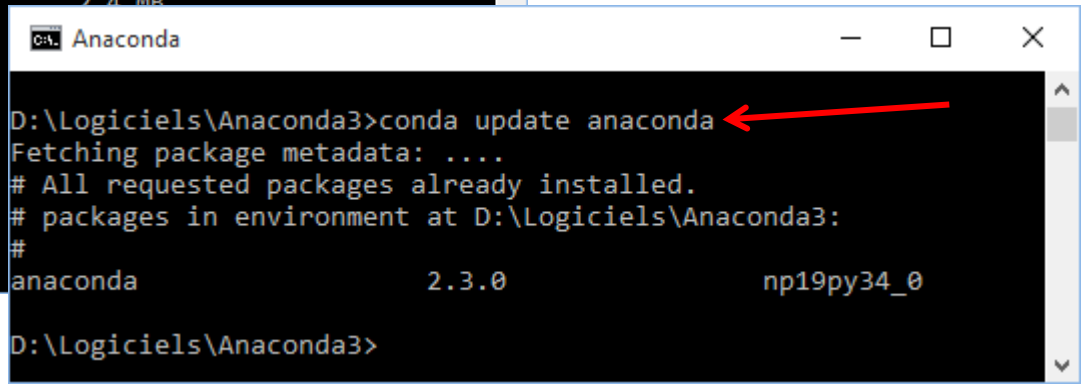
Pour disposer de la toute dernière version, aller dans le terminal DOS (Invite de commande Windows) et insérer les commandes suivantes :

> conda update conda

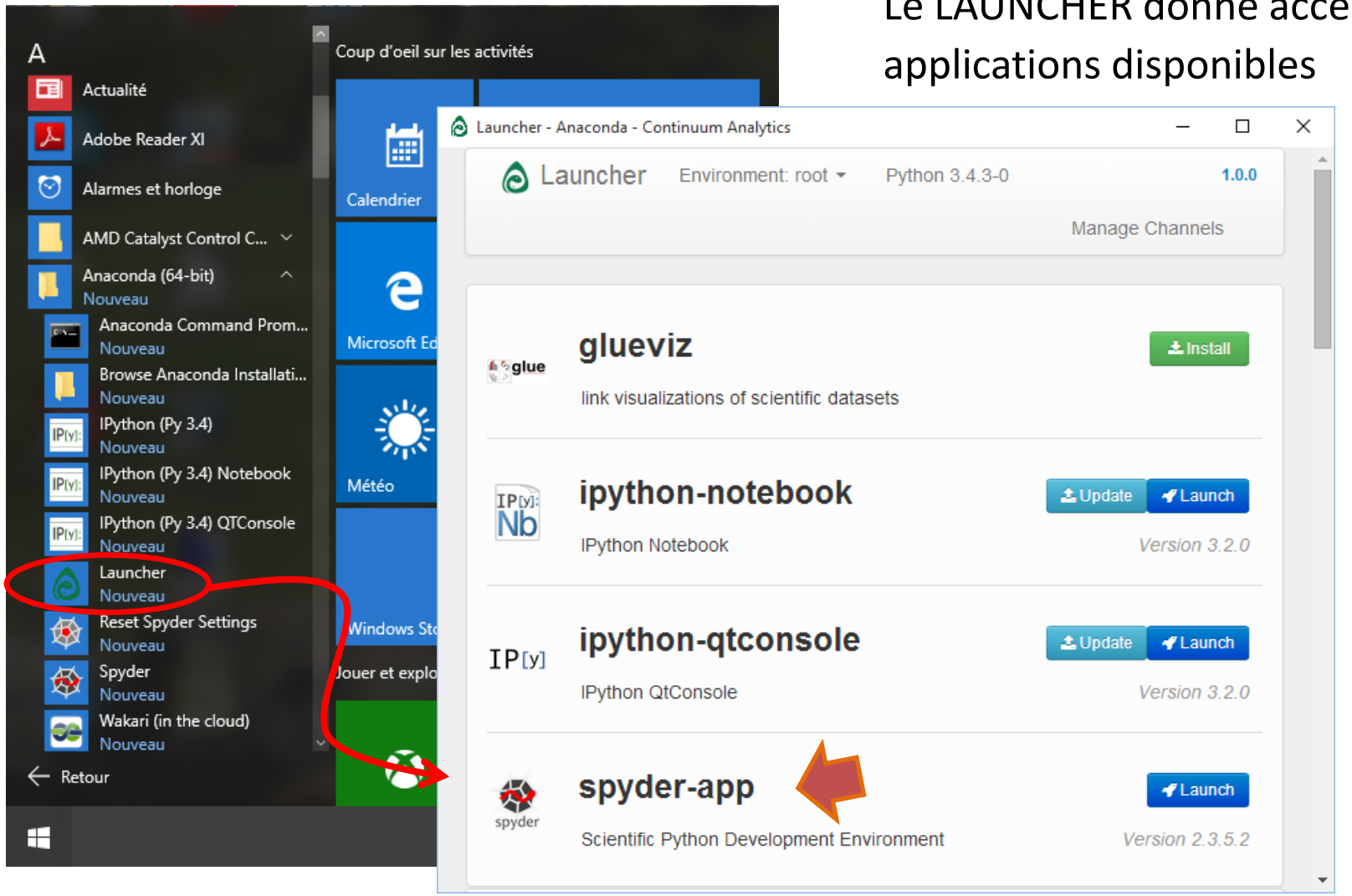
> conda update anaconda

Conda est le gestionnaire de package d'Anaconda.



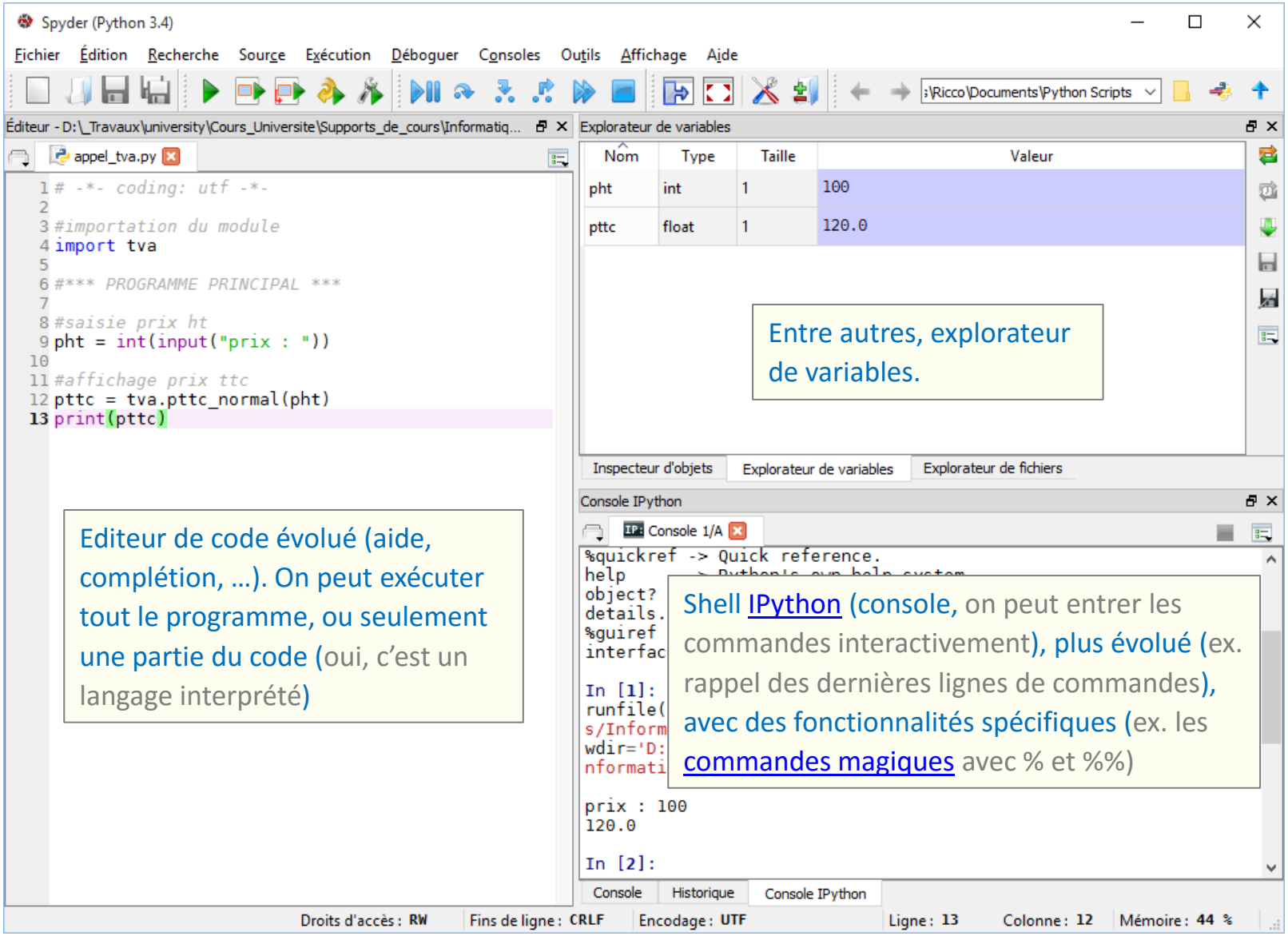


Le LAUNCHER donne accès aux applications disponibles



Sélectionner l'environnement de développement « spyder-app » (cliquer sur Launch).

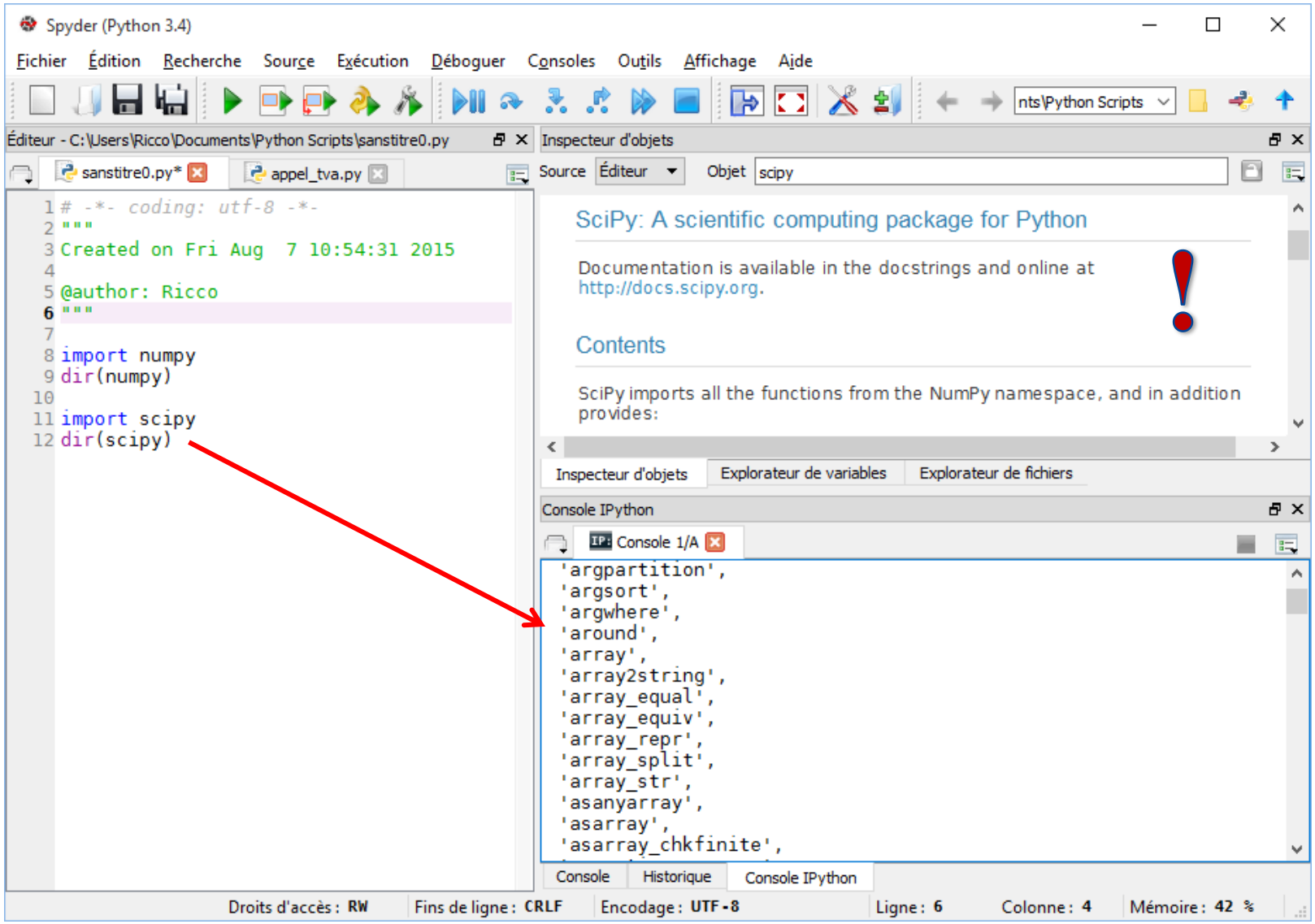
Environnement Spyder



On dispose d'un environnement de développement intégré qui n'est pas sans rappeler RStudio (pour les férus de R)

Et les packages alors ...

Ceux qui nous intéressent – calcul scientifique, calcul statistique – sont directement disponibles!



Références

De la documentation à profusion (**n'achetez pas des livres sur Python**)

Site du cours

http://eric.univ-lyon2.fr/~ricco/cours/cours_programmation_python.html

Site de Python

Welcome to Python - https://www.python.org/

Python **3.4.3** documentation - https://docs.python.org/3/index.html

Portail Python

Page Python de Developpez.com

Quelques cours en ligne

P. Fuchs, P. Poulain, « Cours de Python » sur Developpez.com

G. Swinnen, « Apprendre à programmer avec Python » sur Developpez.com

« Python », Cours interactif sur Codecademy

POLLS (KDnuggets)

Data Mining / Analytics Tools Used

Python, 4ème en 2015

What languages you used for data mining / data science?

Python, 3ème en 2014 (derrière R et SAS)